



Definición del proyecto

Transcripción

[00:00] Hola a todos. Sean bienvenidos a la segunda clase de nuestro curso de introducción a objetos con Java. Ya hemos visto en la clase anterior que orientación a objetos es un nuevo paradigma de programación en el cual buscamos representar digamos el mundo real a nivel ya de programación. Entonces, vamos a partir un poco cuestionándonos: ¿Qué sería en realidad un objeto?

[00:25] Un objeto no es nada más y nada menos que la representación del conjunto de atributos que definen alguna entidad o algún tipo. Por ejemplo, tenemos un auto y el auto tiene diferentes tipos de atributos, como pueden ser marca, color, año, etcétera. Ese conjunto de atributos definen ese tipo de auto. Pueden haber muchos tipos de auto con el mismo conjunto de atributos y en otros casos con más atributos.

[00:58] Por ejemplo, van a ver veces que va a ser importante el tipo de motor o de repente el tipo de caja de cambios, no lo sé. Entonces, la representación de objetos básicamente es eso, es agrupar el conjunto de características de una entidad o de un tipo para que represente sistemáticamente su significado tanto a nivel de atributos como a nivel de acciones. Para entender esto más a fondo y de una forma más didáctica vamos a crear digamos un banco.

[01:34] Nuestro proyecto que vamos a seguir en el curso, va a ser una entidad financiera, va a ser un banco que lo vamos a llamar Byte Bank. Este banco, como todos los demás, va a tener distintos tipos de objetos o de tipos, por así decirlo de alguna manera. Va a tener distintos tipos de entidades interactuan

en él. Por ejemplo una entidad pueden ser los clientes, puede ser la cuenta, puede haber relación entre entidades, que por ejemplo un cliente puede tener muchas cuentas.

[02:08] Y también pueden tener comportamientos u operaciones, por ejemplo la cuenta puede tener digamos retiros, puede tener depósitos, puede tener transferencias. Entonces son comportamientos que son anexados ya a la entidad cuenta.

[02:27] Para comenzar el proyecto, vamos a definir primero qué conjunto de características van a definir nuestra cuenta, para mantenerlo simple, para mantenerlo lo más simple posible nuestra cuenta va a estar definida por cuatro campos que son el saldo, nuestro número de agencia, el número de la cuenta y el nombre de nuestro titular.

[02:47] Quizás en algunos casos o en otros ejemplos puedan ver que una cuenta está compuesta por más atributos, por más campos, como por ejemplo código de país, tipo de moneda, etcétera, pero en este caso, para tenerlo lo más simple posible vamos a tener solamente esos cuatro atributos, que creo que son suficientes para implementar algunas operaciones básicas.

[03:11] Ahora, sabemos que cuenta representa aquella entidad con la que vamos a realizar operaciones pero esto sería hablando en lenguaje humano. ¿Cómo sería representar una cuenta ya en lenguaje de computador, propiamente en lenguaje Java? Para esto vamos a recurrir a nuestro ya conocido editor Eclipse, y bueno, vamos a crear un nuevo proyecto, vamos a repasar un poco lo que ya hicimos en el curso pasado, vamos a new, Java Project y vamos a llamarlo proyecto-bytebank.

[03:55] Vamos a darle finish directamente y él ya creó nuestro nuevo proyecto aquí. Si abrimos aquí vamos a ver que está el folder src tranquilo y aquí tenemos otro proyecto abierto con el que hemos estado trabajando

anteriormente. Yo en este caso lo voy a cerrar. Para eso solamente le damos un clic derecho, close project, y el proyecto ya está cerrado.

[04:19] ¿Qué ganamos cerrando un proyecto? Quizás si este proyecto de aquí tuviera errores de compilación o alguna cosa así, nos estaría saltando un error a cada momento. Y como no vamos a trabajar con este proyecto y no deseo eliminarlo, lo voy a tener cerrado.

[04:37] Volviendo a nuestro proyecto del banco, vamos a crear nuestra primera clase que en este caso va a ser cuenta. Para eso vamos a darle clic derecho, new class, y aquí vamos a darle el nombre cuenta. Vamos a darle finish y vemos que ya creó nuestro primer objeto y vamos a trabajar en base a él.